Detecting Selfish Nodes in MANETs

Bathi Srikanth

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India

Detecting Selfish Nodes in MANETs

Thesis submitted in partial fulfilment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Computer Science)

by

Bathi Srikanth

(Roll No. 212CS1098)

under the supervision of

Prof. M. N. Sahoo

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India

June 2014

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

Certificate

This is to certify that the work in the thesis entitled ***Detecting Selfish Nodes in MANETs*** by ***Bathi Srikanth*** is a record of an original research work carried out by her under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology with the specialization of Computer Science in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Place: NIT Rourkela
Date: June 1, 2014

Prof. M. N. Sahoo
CSE Department
NIT Rourkela, Odisha

Acknowledgment

I am grateful to numerous local and global peers who have contributed towards shaping this thesis. At the outset, I would like to express my sincere thanks to Prof. M. N. Sahoo for his advice during my thesis work. As my supervisor, he has constantly encouraged me to remain focused on achieving my goal. His observations and comments helped me to establish the overall direction of the research and to move forward with investigation in depth. He has helped me greatly and been a source of knowledge.

I extend my thanks to our HOD, Prof. S. K. Rath for his valuable advices and encouragement.

I would also like to thank specially Saritha (Sister) and friends for standing besides me all the time and support me morally and ethically.

I must acknowledge the academic resources that I have got from NIT Rourkela. I would like to thank administrative and technical staff members of the Department who have been kind enough to advise and help in their respective roles.

Last, but not the least, I would like to dedicate this thesis to my family, for their love, patience, and understanding.

Bathi Srikanth
Roll: 212CS1098

Author's Declaration

I hereby declare that all work contained in this report is my own work unless otherwise acknowledged. Also, all of my work has not been submitted for any academic degree. All sources of quoted information has been acknowledged by means of appropriate reference.

Bathi Srikanth

Roll: 212CS1098

Department of Computer Science

Abstract

Mobile Ad-hoc Networks(MANETs) is self configured and decentralized wireless network without any prior infrastructure. Every node in it acts as a router as well as end-system and hence each node in MANET is allowed to move freely which makes routing difficult. Most of the MANET routing algorithms like AODV and DSR assumes that every node will forward every packet it receives. Source node will relay packets to the destination node through the intermediate nodes. However, misbehaviour of the selfish nodes is a common phenomenon in MANET. These nodes use the network and its services and do not provide any services to intermediate nodes in order to save energy such as battery, CPU Power and bandwidth for relaying data from other nodes and reserve for themselves. These selfish nodes will degrade the performances of wireless ad hoc networks. However, we can identify the selfish nodes by modifying the original AODV and DSR routing algorithms. In this thesis, we proposed a time based scheme for identifying selfish nodes and perform the simulation using Network Simulator 2.34.

Keywords: *MANETs, Selfish nodes, AODV*

Contents

List of Figures

List of Tables

Chapter 1

Chapter 1

Introduction

1.1 Introduction

Ad hoc network refers to a network connection built for a single session and does not require a wireless base station and a router, it is a temporary network association made for some particular reason like for sending date from one device to other. If the network is set up for a long period of time, then it is just a plain old local area network.

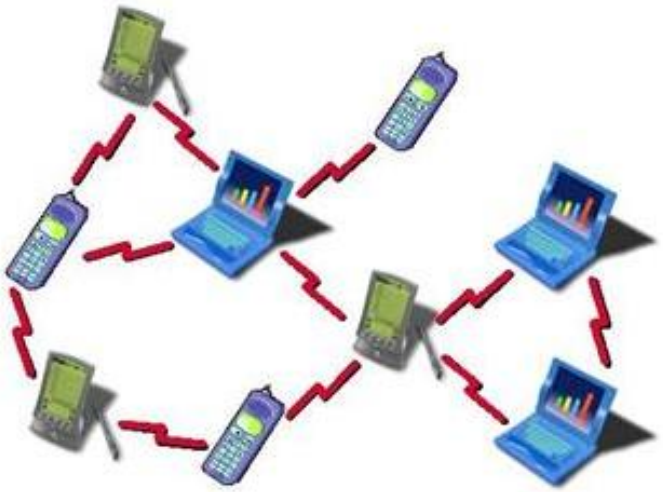

Figure 1.1: Ad hoc Network

1.2 Wireless Ad hoc Network

The mobile adhoc network is an integration of more than one wirelass nodes and have the capacity of transfering data to one another without any kind of help

from a centralized administrator. Every device acts as a router and end system in adhoc network. The network topology in a wireless adhoc network is dynamic due to the integration of the nodes changing with time because of the mobility of nodes, entry of new nodes and flight of nodes. Hence, a productive routing protocols are needed for these nodes to communicate.

Quick and unusual topological changes, wireless network dynamic nature, mobility of nodes and restricted battery power raise numerous difficulties in making up a routing protocol. Because of huge challenges in planning a routing protocol for MANET, various developments recently focusing on giving ideal solution for routing. Thus, an ideal routing protocol that can cover the greater part of the user requirements or applications and additionally adapt upto the stringent conduct of the wireless medium is constantly alluring.

Ad hoc nodes are devices to have the capacity to identify the existence of other such devices in order to permit data sharing and communication. Besides that, it ought to additionally have the capacity to distinguish type of relating attributes and services. Due to the mobility of nodes the amount of wireless nodes will change, routing data additionally changes to follow changes in the connectivity of links. Henceforth, the topology of the network is a great deal are dynamic and the adjustments are frequently unusual as contrasted with the settled type of actual wired networks.

1.3 Characteristics of Ad hoc Network

For designing or suggesting solutions for MANETs following features can be considered

- Distributed operation is one of the feature of MANET because in ad hoc network every device works individually and there is no machine or centralized administrator to deal with this network, insted this job is conveyed among all working nodes. Every device works with an alternate device in collaboration to accomplish functions like routing and security.
- As compared to wired network MANET has lower bandwidth capacity.

MANETs can encounter an issue of lower bandwidth capacity and bit error rate because node to node link path are utilized by many nodes in the network.

- An alternate characteristic of MANET that could be utilized is energy as a part of mobile nodes. As all mobile nodes will get their energy from the battery, which is a constrained asset, what ever energy the portable nodes have, it must be utilized proficiently.

- Another characteristic of MANET is security. Because the information and devices in MANETs are insecure from threat, it is the most important concern in this network. The main threats to security are denial of service attack, spoofing, and eavesdropping.

- Nodes in MANET alter their positions arbitrarily as they are allowed to move anywhere and these mobility of nodes causes frequent disconnections. Hence, the network topology in MANETs will always change and so the dynamic topology should be supported by the nodes in MANET.

- A MANET incorporates a few focal points over wireless networks, in addition to simplicity of arrangement, rate of organization, and expanded dependencies on a settled base. A MANET is attractive because of the fact that it gives an immediate network accumulation without the existence of base station and framework organization.

1.4 Classifications of Ad hoc Network

Ad hoc networks can be classified on the essence of the network size, node configuration, topology and the communication procedure(multihop/singlehop).

1.4.1 Classification based on the communication

In ad hoc networks communication can be either multihop or singlehop, depending on the configuration.

Singlehop ad hoc network In single hop network all the devices which are in the communication range can communicate directly without the aid of any other devices. All these nodes are dynamic, however they must be in the communication energy of all nodes, which tells that whole network moves as a group.

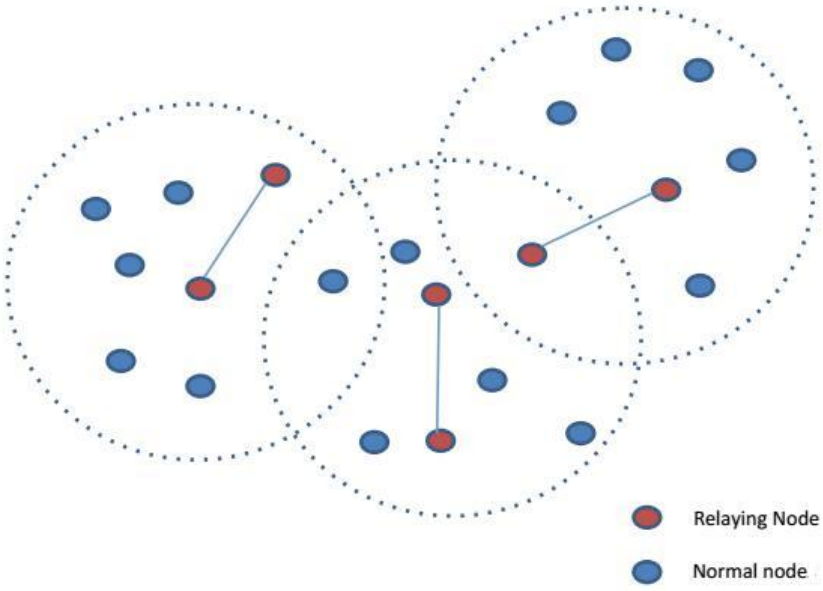

Figure 1.2: Single-hop Ad hoc network

Multihop ad hoc network In multihop network intermediate nodes can help in communicating if the nodes are out of communication range. The traffic of these end nodes are forwarded through some intermediate nodes. The difficulty of the network is the mobility of nodes where by the topology of the network alters continuously. Assigning a routing protocol is the general problem in this network.

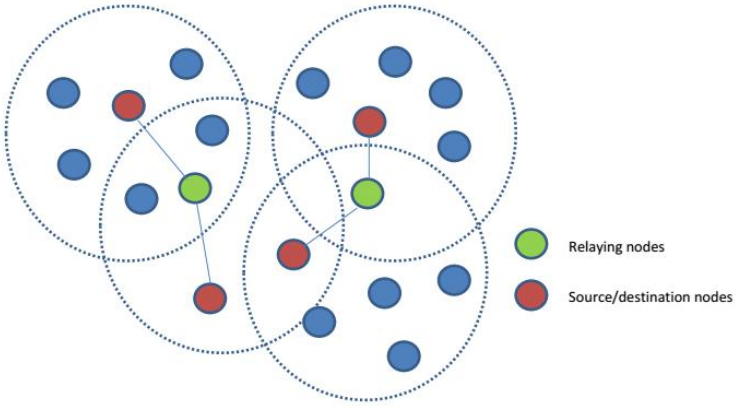

Figure 1.3: Multi-hop Ad hoc network

1.4.2 Classification based on the node configuration

Based on node configuration of the hardware, ad hoc network is further classified.

Homogeneous Ad hoc Network In this network, all nodes have the same qualities, seeing the hardware setup as peripheral devices, display, memory and processor. Most will know wireless sensor network is the representation of homogeneous network.

Heterogeneous Ad hoc Network In this network, the nodes contrast as per the hardware configuration. Each node has distinctive qualities, assets and arrangements. In this kind of ad hoc network all the nodes do not provide same kind of services.

1.4.3 Classification based on the topology

Based on the topology ad hoc network is classified. Every single node in an adhoc network are divided with specific functions such as aggregate, hierarchical and flat ad hoc network.

Flat Ad hoc Network In this network, there is no difference between every single node, all nodes convey same responsibility. All nodes are equivalent the ad hoc network. Throughout out the network globally the control messages have to

be transferred, but they are suitable for exceedingly dynamic network topology. The adaptability diminishes when the amount of nodes expands fundamentally.

Hierarchical Ad hoc Network This kind of network comprises of many clusters, every cluster is considered as a network and all they are connected together. The nodes in hierarchical network can be ordered into two sorts.
Normal node: These nodes communicate directly with in the cluster and communicates with nodes in other cluster with the help of the master node.
Master node: These nodes administrate the cluster and is responsible for transferring data to another cluster.

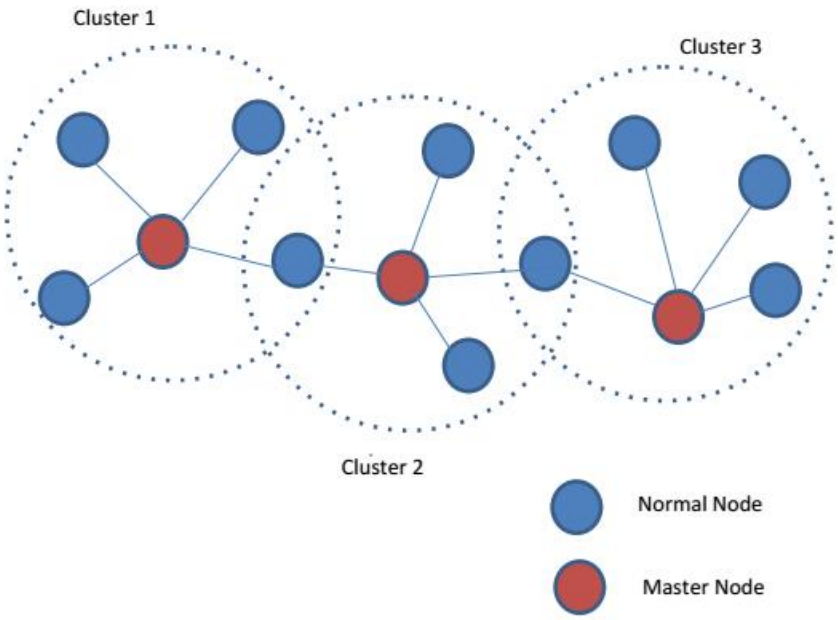

Figure 1.4: Hierarchical Ad hoc network

1.5 Applications of MANET

With the increse in the portable devices and in addition advance in wireless communication, ad hoc networking is picking up imperativeness as a result of its growing number of far reaching applications. Ad hoc networking could be connected anyplace at any time without any framework and its adaptable networks.

Ad hoc networking permits the nodes to keep up connections to the network and in addition effectively adds and expels nodes in and out from the network.

Emergency services

- For replacing fixed infrastucture in case of ecological disasters
- Fire fighting and policing
- Emergency rescue operation
- Support in hospital for nurses and doctors

Sensor Networks

- Body area networks (BAN)
- For tracking movements of animals, detection of biological/chemical environment conditions.

Enterprise and home networking

- Office/home wireless networking .
- Meeting rooms, coference hall, personal area networks (PAN), personal networks (PN).
- Construction site networks.

Tactical networks

- Military objects moving at high speed such as tanks, warships and airplanes.
- Battle fields which are automated.

Commercial, civilian and education environments

- Networks of visitors at airports .
- Virtual classrooms, universities and campus settings.
- Ad hoc communications during meetings or lectures .

1.6 Organisation of thesis

The thesis is organized as follows: Chapter-2 describes the background concepts related to our work. Chapter-3 discusses the literature surveys that have been done during the research work. In Chapter-4 the proposed time based scheme for detecting selfish nodes in MANETs is described. In Chapter-5 simulation results and analysis were discussed for the proposed algorithm. Finally in Chapter-6 concludes with the summary of work done.

Chapter 2

Chapter 2

Preliminaries

For understanding the meaning of selfish node and the working of the AODV routing protocol, misbehaviour of selfish nodes and ad hoc on demand distance vector(AODV) routing protocol is introduced in this section.

2.1 AODV Routing Protocol

With AODV algorithm multihop, self starting, dynamic routing can be enabled between the mobile nodes that wish to maintain and establish an ad hoc network. It permits and helps mobile nodes in acquiring routes rapidly for new destinations, and does not oblige devices to keep up routes to destinations that are not in dynamic communication. This protocol enables mobile devices to react to the changes in network topology and link breakages in a timely and efficient way. In case if a link breaks, AODV helps in notifying the set of nodes that are affected so that the routes using the lost link can be invalidated.

In AODV four control messages are defined for maintaining routes to the destination. These contol messages [16] include RREQ(RouteRequest) message, Hello message, RERR(RouteError) message and RREP(RouteReply). Periodically a hello message is broadcasted by every node in the network to all its neighbors to tell that it is alive. Whenever a neighboring node receives a hello message, the neighbor node includes the data about the node which sends a hello message into its routing table.

If a node want to communicate with some other node, the source node will

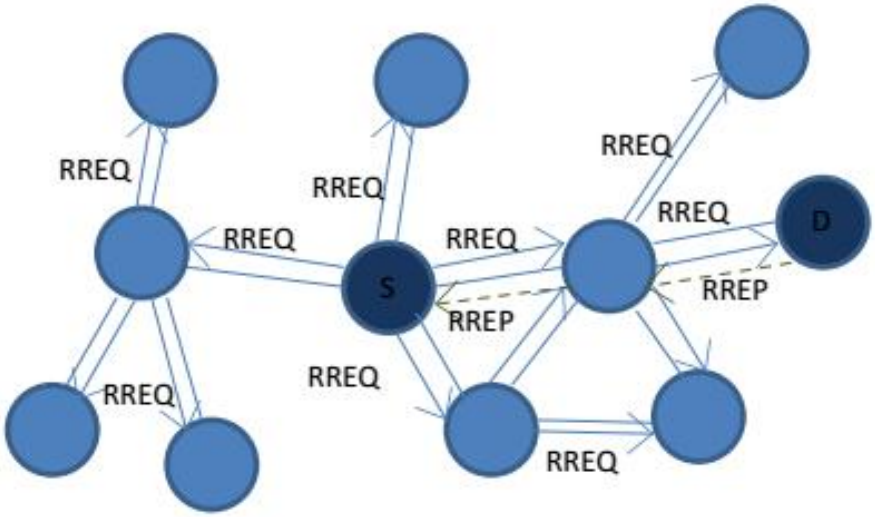

Figure 2.1: AODV Routing

check destination node in its routing table. Route request(RREQ) packet is broadcasted by the source node to all its neighbors in case if the routing table does not contain destination node [1]. Every neighboring node likewise rebroadcasts the gained route request(RREQ) messages to its neighbors. Through along these lines over and over until the destination node is reached. If the neighbor node accepts the route reply packet(RREP), it likewise replies conversely the Route reply packet to the former neighbor node as per the data in its routing table. The transmission path can be created at the point when the route reply(RREP) message is sent again to the originating node. Throughout the information transmission, if in this transmission way a node is not able to communicate with the neighbor nodes, then a route error(RERR) message is sent by this node to the source node and the data that belongs to this transmission way is deleted from its routing table. The source node will retransmit RREQ packet for building a new transmission path when it receives a route error(RRER) message considering that the transmission path to the desired destination node has broken.

2.2 Selfish Node Behaviours

Selfish nodes are inclined to get the greatest profits from the networks and at the same time these nodes trying to conserve their own resources like bandwidth, [1] batterylife or hardware. A selfish node only communicates to other nodes if its data packet is required to send to some other node and refuses to cooperate other nodes whenever it some data packets or routing packets are received by it that it has no interest in. Hence data packets are either refused to retransmit or are dropped for being received by a selfish node. The selfish nodes behaviors in AODV routing protocols can be as follows:

2.2.1 Nodes which do not send Hello packet:

The principle target of this sort of selfish node is hiding itself and to abstain from being included in the others transmission way.

2.2.2 Nodes which do not forward RREP messages:

Because of this kind of selfish behavior whole network will be paralyzed. In AODV, the source node will get a RREP message from the destination node through some intermediate nodes to establish a complete transmission path, but here the communication path will not be established because this kind of selfish nodes will not forward the RREP message. Hence the source node will broadcast Route Request(RREQ) message continuously.

2.2.3 Nodes which do not forward Data messages:

The misbehaviour of this type of selfish node impacts the performance of MANET by dropping all the data messages that are received by these nodes. Instead of relaying these data messages these will be dropped.

2.2.4 Nodes forwarding RREQ messages with delay:

When this kind of selfish node gets a Route Request(RREQ) message it forwards this RREQ message after some lag near the upper bound of time out for not to

participate in a route.

2.2.5 Nodes which do not forward RREQ messages :

In MANET, if this type of selfish nodes receives some RREQ messages, then instead of forwarding these RREQ messages, these messages are dropped and thus these kind of selfish nodes skips being the route member for other nodes. Thus avoiding forwarding these messages for others as a result more nodes are required for building a transmission path.

2.2.6 Selfish Behaviour Depending on the Nodes Energy:

This type of selfish nodes act normally if its energy level lies between full energy level and some threshold k1. They act like do not forward data messages selfish node if its energy level lies between k1 and some k2 and if its energy level is below k2 then they behave like do not forward RREQ message selfish node.

Chapter 3

Chapter 3

Litreture Review

The techniques for detecting or preventing selfish nodes in the network are

3.1 Watchdog Method

It is a scheme [9] for selfish node detection in MANET by overhearing other nodes. A buffer is maintained by each node for the packets sent recently and the packets within the buffer are compared with overhearing packet to check if there is a duplicate. Then the packet in the buffer is discarded and blank out by the watchdog. If the packet has stayed longer than a certain time-out in the buffer, then the watchdog will increase the fault count for the node culpable for sending the packet. If the count crosses some threshold, the node is considered to be misbehaving and a message about this node is sent to the source.

Total number of packets incoming are equal to total number of packets outgoing in watchdog. Watchdog is presented in every node in the network. In the following Fig 3.1. Node S is a source and node D is a destination. Node S forward the packets to node Watchdog present in node S overhears the neighbor node A whether it forward the packets to neighbor node B. Here node A forward the packets to node B. Similarly, watchdog present in node A overhears whether node B forward the packets to node D. The problem with watchdog is partial dropping, false misbehavior, limited transmission power, receiver collisions and ambiguous collisions might not be detected.

Figure 3.1: Watchdog

Pathrater In the path rater includes the knowledge of link reliability data and misbehaving nodes to find the most reliable route. Every node in the network maintains a metric for all the nodes it knows about. It measures a metric for path by balancing the node ratings in the route. Path with higher rating is chosen if multiple paths are there to same destination.

3.2 2ACK Method

The 2ACK scheme [2] [21] is used for detecting misbehaving link rather than detecting selfish nodes. For the existing routing protocols like DSR it can be used as an add-on. A fixed route of 2 hops(3 nodes) in the direction that is opposite to the direction of data traffic is assigned to a 2Ack packet

At whatever point a route must be framed from the source to the destination, we first utilize the essential directing protocol like DSR. To apply the 2ACK strategy, we picture the whole route as set of sequential covering triplets.

For example, if 1-A-B-C-D-E-2 represents a route from source to destination, then the 2Ack technique is applied to every triplet of the set: (1, A, B) (A, B, C) (B, C, D) (C, D, E) (D, E,2). Working of the technique shown below:

We consider triplet(A,B,C) for which the algorithm is applied, A sends a data packet to node B which has to be forwarded to node C along the route. Node A must be guaranteed of the effective gathering of the packet by node C through the acknowledgement packet 2Ack from C to B and from B to A. As such a reverse 2hop route is followed by 2Ack packet. The node C in the triplet is called 2Ack sender and node A is called 2Ack receiver. The timely and successful entry of 2Ack packets for each transmission guarantee node A that the link B-C is working well and not misbehaving.

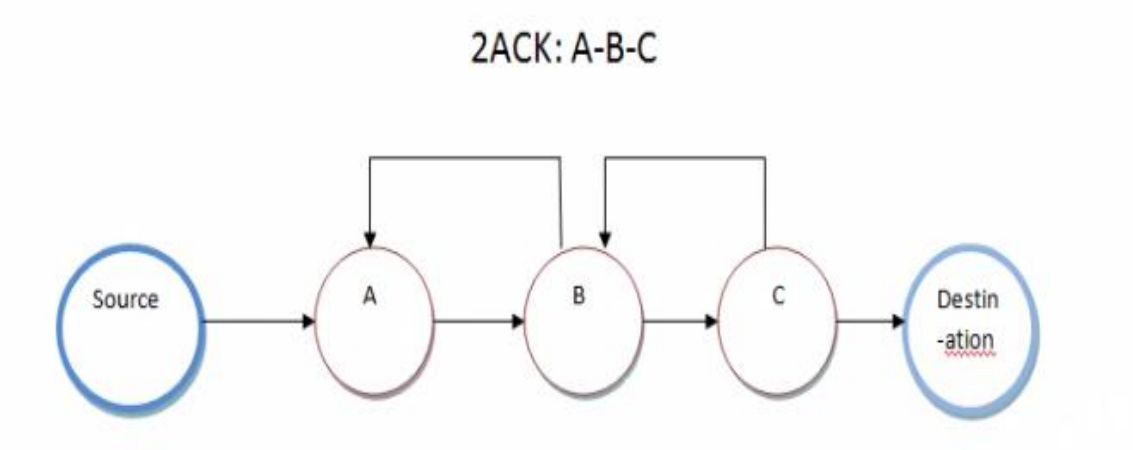

Figure 3.2: 2ACK

For all the triplets in the path 2Ack transmission takes place. Hence, the destination and just last node before the destination will not serve as a 2Ack receiver and very next router to the source node will not act as a 2Ack sender. Only some chunk of data packets is acknowledged for reducing the additional overhead in routing.

3.3 A Distributed Approach for Detecting and Deleting Selfish nodes

The data processing and gathering module of the framework gather information in two ways [6], first it generally runs an observing methodology to get the conduct data of neighbor nodes and besides it trades this data with different nodes checked data.

A rating count is generated for each node after processing data collected by gathering data and processing module. Rating count of a mobile device is nothing but the conduct of the device checked by other devices in the system. By having the information about rating count the mobile devices are ordered to the cooperative and non- cooperative in the collaborative decision making module. Based on the decision that is given by the decision module, action is performed by the response operation module. Misbehaving nodes are ignored in routing operation.

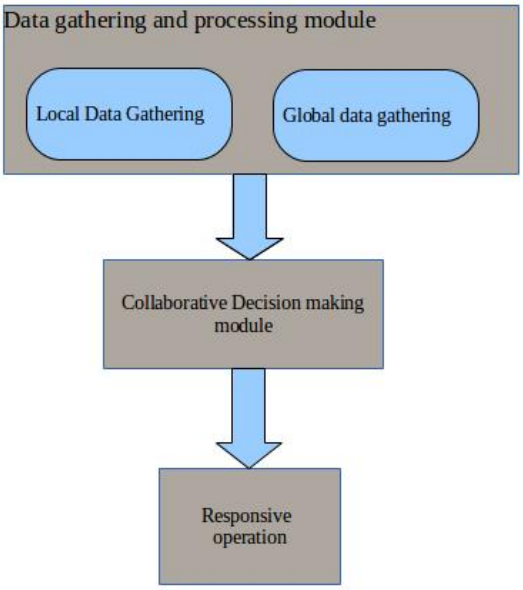

Figure 3.3: A Distributive approach

3.4 A Reputation-Based Scheme to enforce Cooperation in MANET

This scheme encourages selfish nodes to cooperate and detect them in wireless ad hoc network. The system [5] will encourage by provide quick service to the cooperating nodes. Three main modules are there in this mechanism for detecting selfish nodes are priority processing system, reputation system and checking system.

3.4.1 Checking system

Every node in this system has a watchdog module and its task is to check the neighboring nodes and observe their behaviours. Firstly all the first hop neighbors are checked by each node and then the count of messages that are received and sent by the nodes are saved and then these are sent to the reputation system. In a specific time period, saved information is updated by this module.

3.4.2 Reputation System:

This system utilizes the extent of the amount of messages which are transmitted by a node to the amount of messages which are gained by a node as the nodes cooperation coefficient. This reputation is the same as coefficient and expressed as follow:

Collaboration coefficient T is figured as, the degree between the amount of packets sent to the amount of packets gained.

A table for reputation is maintained in each node to monitor and check all its one hop neighbors. The range of the value 'T' will be in between zero and one. The cooperation of the node will be low if the value of that node in its reputation table is near to zero and these nodes are considered as selfish nodes, if the value is near to one then these nodes reputation is high and considered as cooperative. In this methodology, as opposed to sending an excess of packets about the reputation of nodes and likewise sending the cautioning packets about the shelfish nodes, including another field to the RREQ packet and embeddings the cooperating coefficient in it, the count of packets will be diminished significantly. In this activity, every originating node put an essential value as the default coopeating coefficient in this field and after ward it will transmit the RREQ packet. The node, which gets this message, examin the source address field of the message. If the source address is in the reputation table then cooperation coefficient of the source node is replaced with its original value.

3.4.3 Priority Processing System:

Relying on the cooperating coefficient the priority of each packet is decided by the priority module. Whenever multiple packets are received by a node and if there is no possibility of concurrent packet forwarding, the cooperation coefficient of the packets of different nodes is checked and the packet with highest coefficient is forwarded first. This technique encourages the cooperating nodes as the services are received by these nodes first and the selfish nodes are penalized by getting services later.

3.5 Credit based scheme

The fundamental thought of credit-based plans [15] are to give motivating forces for nodes to reliably perform functions of the network. With a specific end goal to accomplish this objective, nugglets a virtual currency scheme can be set up. Node forwards others traffic will get paid. When they demand different nodes to help them for message sending, they utilize the same payment scheme to pay for such administrations. Implementation of CBS can be done in two ways

- Packet trade model(PTD)
- Packet purse model(PPM)

3.5.1 Packet trade model(PTD)

- The node trade for beans by intermediate nodes for packet forwarding.
- Every midway node purchases packet from the past node for some nuggets and offers it to the following node for few nuggets.
- The destination of the message covers total cost for transmitting the packet.
- There is no need for the source node to know in advance the number of beans required to send the packet to the destination node.
- The cost of transmitting a packet is high from the originator to destination.

3.5.2 Packet purse model(PPM)

- In this scheme for packet forward service the source node of the packet will pay.
- Among all the forwarding nodes the service charge is distributed. The source node fill with sufficient amount of beans to arrive at the destination.
- One or more nuggets will be acquired from the packet by each node. In PPM each forwarding node gets some nuggets from packet.
- The packet is discarded. If the packet does not have enough nuggets to forward.
- The problem with PPM is the difficulty in estimating the number of nuggets that are needed to forward a packet to the destination.

3.6 Motivation

Every node in MANET relays its traffic through some intermediate nodes. Since the mobile nodes are commonly bound by computing and power resources, some nodes refuse to cooperate, this kind of misbehaviour impacts the fairness, reliability and efficiency in MANET. We were motivated to detect this kind of nodes which are not cooperative in order to improve the performance of the network.

3.7 Objective

- Objective is to design an efficient algorithm to increase the True Detection Rate(TDR) and to reduce the False Detection Rate(FDT) of selfish nodes in MANET.
- To analyze and validate the performance with the help of Network Simulator 2.34.

Chapter 4

Chapter 4

Proposed Method

4.1 Introduction

In AODV, every selfish device merely means to spare its own resources for itself, it is simple for the node to turn into a selfish node overlook all messages(control and data) which are not intended to it. The nodes which don't send RREQ packets don't impact the network, this sort of selfish nodes can increase end to end delay because the number of nodes in the transmission path will increase.

In AODV routing protocol, a hello message is sent to obtain the neighbors information. Connectivity can be determined [20] by two variables using hello messages. ALLOWED HELLO LOSS and HELLO INTERVAL. Duration between the two hello messages of a node is known as the HELLO INTERVAL.

ALLOWED HELLO LOSS points out the greatest number of times of HELLO INTERVAL to hold up without getting a hello message before discovering a loss of connection to a neighbor. The prescribed worth for ALLOWED HELLO LOSS is two seconds and for HELLO INTERVAL is one. As it were, if a hello message is not accepted from a neighbor inside two seconds of the last message, connectivity lost is determined to that neighbor node.

4.2 Proposed methodology

In my proposal, every checking node works in promiscuous mode and might monitor the neighboring nodes which don't forward RREQ packet. Every checking node will maintain an entry for each of its neighboring nodes. In original AODV each node will contain the neighbor node address and the neighbor node expire time, newly added fields in the neighboring table are

- last_helloTimer
- last_serviceTimer
- node status

Neighbor nodes last service timer : Last service time is the time in which last time the neighboring nodes provided service to the network, providing services includes sending/forwarding RREQ packets, sending RREP/RRER and data packets .

Neighbor nodes last hello timer : Last hello time is the time recorded when the neighbor node has last sent the hello packet.

Neighbor nodes status : Status is the neighbor nodes current behavior recorded. Initially status of the neighbor nodes is initialized to zero, which is the behavior of the node is unknown .

neighborNode_addr	neigborNode_expire	last_HelloTimer	last_serviceTimer	status

Table 4.1: Neighboring node table fields

The two fields last_ServiceTimer and last_HelloTimer are updated for every action performed. If the difference between the neighbor nodes last_HelloTimer and the last_ServiceTimer is with in some threshold, then the node is considered as normal.

We call the threshold as action holdoff time . If the difference between the two timers exceed some threshold then the node is suspected as selfish and further testing is done because some nodes can be falsely identified as selfish.

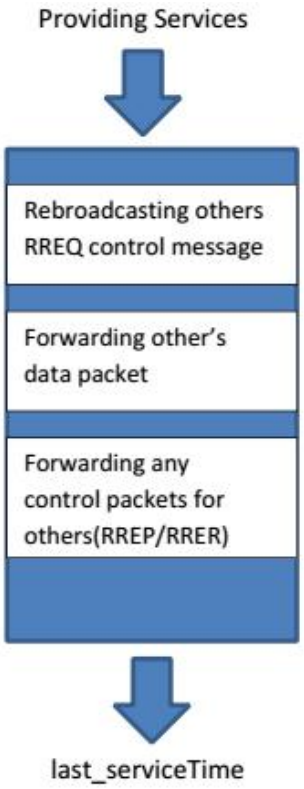

Figure 4.1: Neighbor node services

To reduce flooding the node that receives a request for forwarding the data packet will perform further testing and allowed to broadcast a fake RREQ message.

When a node gets a request for forwarding data message first it will examin the contrast between nodes last_hellotimer and last service timer. If the difference exceeds some threshold, then the checking node will send a fake RREQ packet through one hop(TTL=1). The checking nodes will wait for this doubtful node to rebroadcast the Route request message before some timeout. If the suspected node react, then the last service timer is updated and the node is considered as well behaved.

If it drops or do not react, then the checking node will mark the doubtful node as selfish. In this proposed technique, every checking node will only regard its own data and will not claim with others, which removes false parsing and false accusation attacks.

4.2.1 Step by step procedure

STEP 1: If a monitoring node hears a neighboring nodes data packet to forward it will check the difference between the last_helloTimer and last_serviceTimer.

STEP 2:

IF The difference between the timers is with in the threshold

(last_helloTimer - last_serviceTimet $\leq$ thresholdα)

THEN The node is considered as normal and the last service time is updated (last_serviceTime = CURRENT_TIME).

ELSE The node is considered as suspecious node and further testing is conducted.

STEP 3: The monitoring node will broadcast a fake RREQ packet(with TTL=1 to reduce flooding) and waits for the doubtful node to rebroadcast the Route Request message before time out.

STEP 4:

IF The suspicious node responds before time out

THEN the last service timer (last_serviceTimer = CURRENT_TIME) is updated and labled as normal node.

ELSE The suspicious node is labeled as selfish node(status = selfish).

Chapter 5

Chapter 5

Simulation Results and Analysis

5.1 Simulation

We have used Network Simulator-2.34 to simulate our proposed scheme.

5.1.1 Simulation Evironment

Area Nodes inner the network range move according to random waypoint mobility model. In this model, every node moves randomly with in the specified network area. Simulation of our proposal was done with in a network size of 1060m x 800m and 25 nodes.

Pause Time The node will remain for some time (pause time) after it reaches to its target location and before going to another arbitrary location. In my proposal the pause time is 0.2seconds.

Traffic Type The communicatin traffic utilized are consistent bit rate (CBR) association with an information rate of 10 pack for every second. 25 associations are created at arbitrary so that every node has opportunity to join with each other node.
Our scheme was simulated with 5, 9, 15 and 21 selfish nodes in the network. For different configuration we evaluated the proposed scheme with different moving rate of the nodes and for different thresholds(Action Holdoff Time).

Area	1060 x 800
Traffic Type	CBR(10packets/s)
Total Number of Node	25
Pause Time	0.2s
Moving rate of the nodes	0m/s, 10m/s, 20m/s
Number of selfish Nodes	10,15,18,21
Routing Protocol	AODV
Simulation time	100s

Table 5.1: Simulation Parameters

5.1.2 Simulation Metrics

True Detection Rate(DR) is the ratio between the count of selfish nodes identified to the aggregate count of selfish nodes in the network.

TDR = Nsd /Nts

False Detection Rate(FDR) is the ratio between the aggregate count of ordinary nodes falsely identified as selfish nodes by more than one ordinary nodes to the aggregate count of ordinary nodes in the network.

FDR = Nmd /Nns

Nsd is the aggregate count of selfish nodes identified by more than one normal nodes in the network.

Nts is the aggregate count of selfish nodes in the network.

Nmd is the aggregate count of behaving nodes identified as selfish by more than one normal node.

Nns aggregate number of ordinary nodes in the network.

Action Holdoff Time is the time in which no action is performed

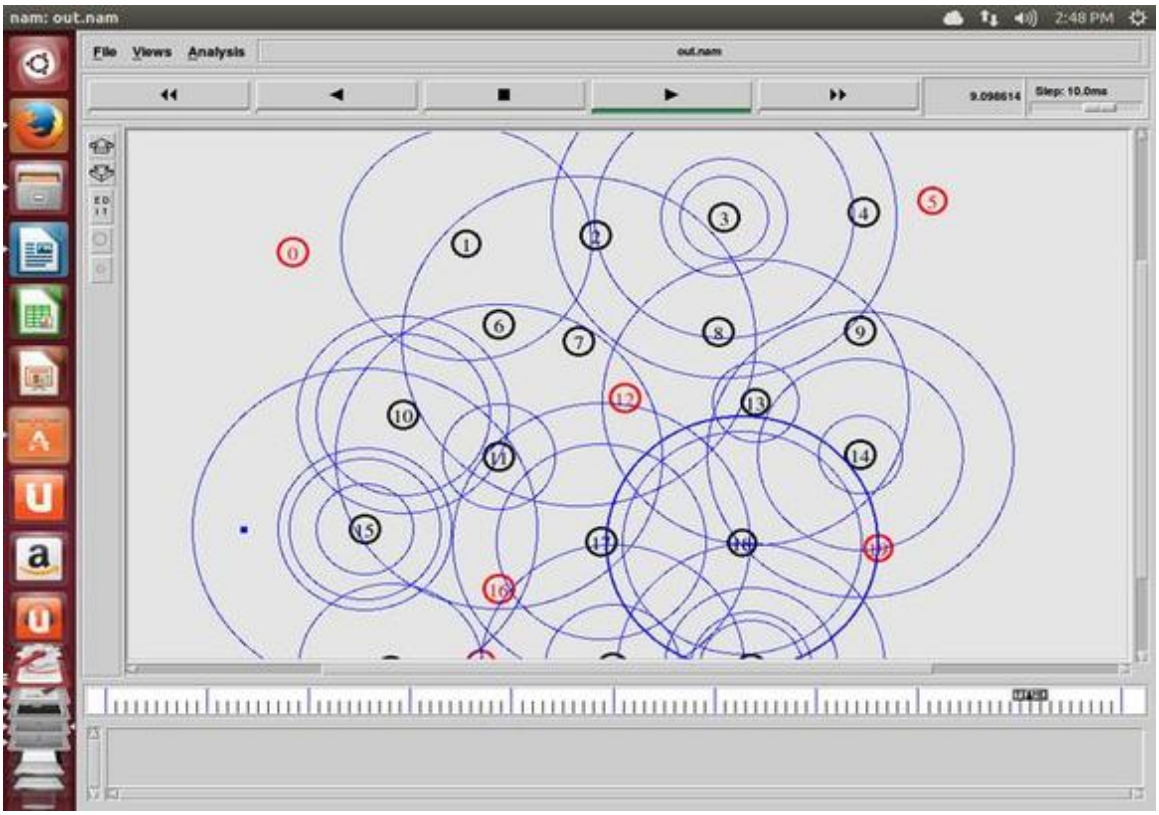

Figure 5.1: Simulation with selfish nodes in the network

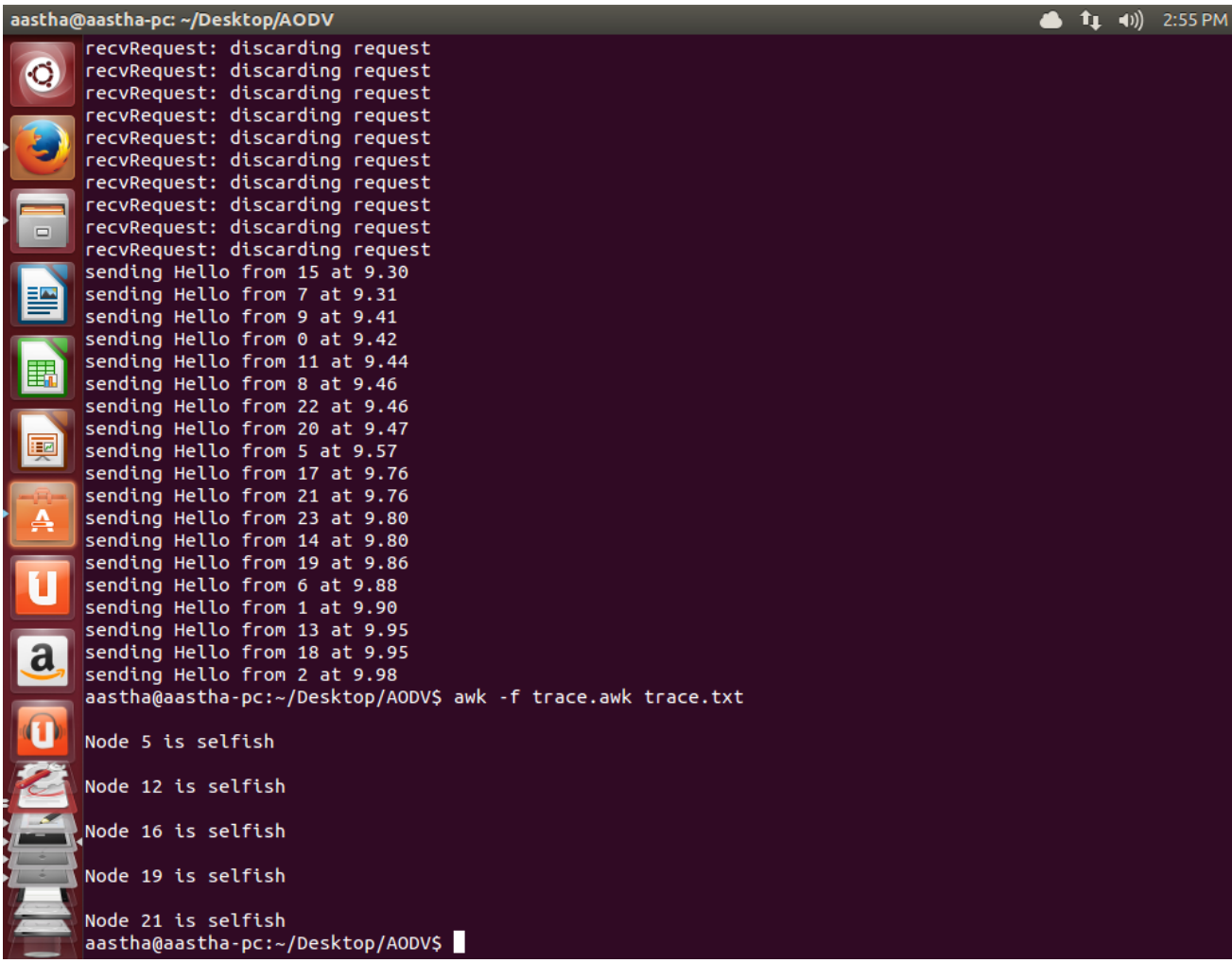

Figure 5.2: Detection Rate of selfish nodes with different moving rates

5.2 Results

From Figure 5.3, We notice that when the count of selfish nodes that don't transmit others route request packets are more then the TDR is less this is because when this kind of nodes are more in MANET, then most of the neighbor nodes will be selfish, and the normal nodes which are in the range of these selfish nodes cannot be identified. Hence, this will lessen the TDR of selfish nodes in the network.

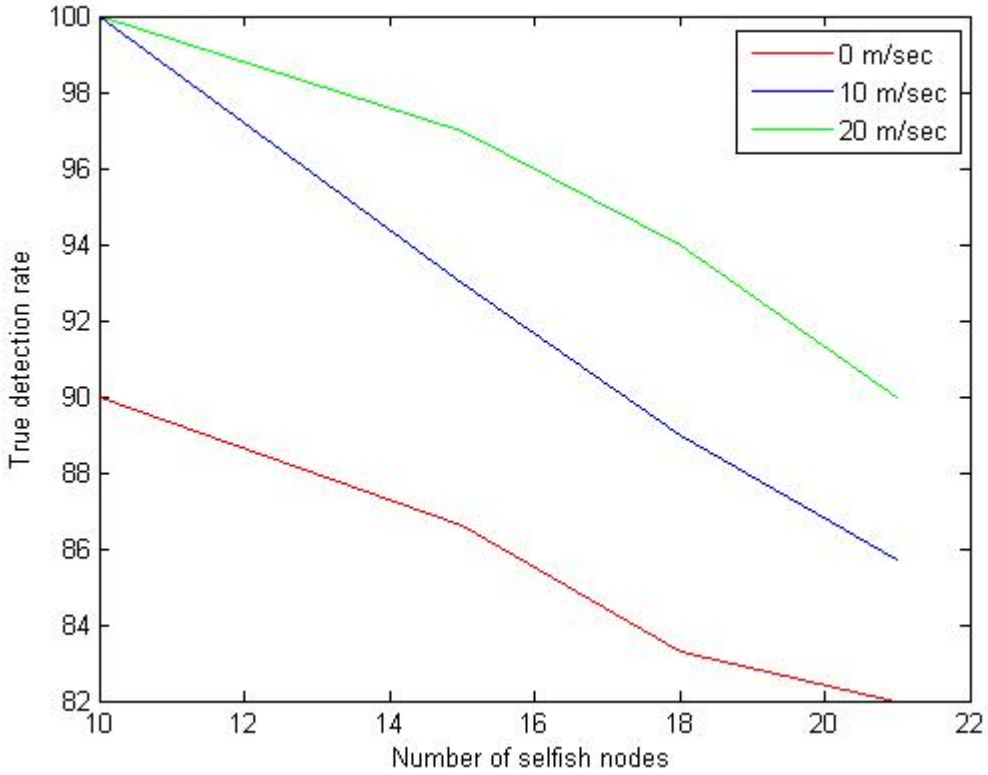

Figure 5.3: True detection rate of selfish nodes with different moving rates

From the Figure 5.4, we notice the FDR of the selfish nodes is high when the mobility rate of the nodes is high, this is because when a node broadcast a packtet to its neighboring node, just in time the neighbor node may go out of communication range and that node will be falsely identified as selfish nodes.

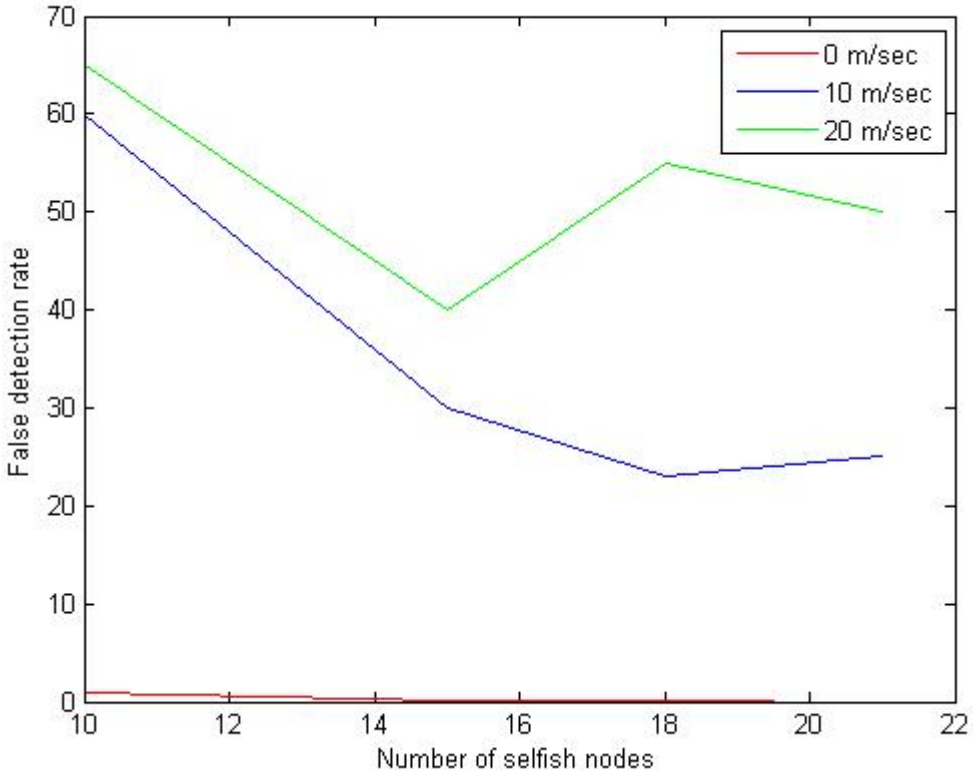

Figure 5.4: FDR of selfish nodes with different moving rates

We also observed the TDR and FDR of selfish nodes with different action holdoff times.

From the Figure 5.5 and Figure 5.6 we notice that if the action holdoff time is less, the true detection rate is high, this is because if the action holdoff time is less then the monitoring of the neighbor nodes will be done more number of times and on the other hand false detection rate increses with the decrease in action holdoff time. Lesser the action holdoff time worse the false detection rate.

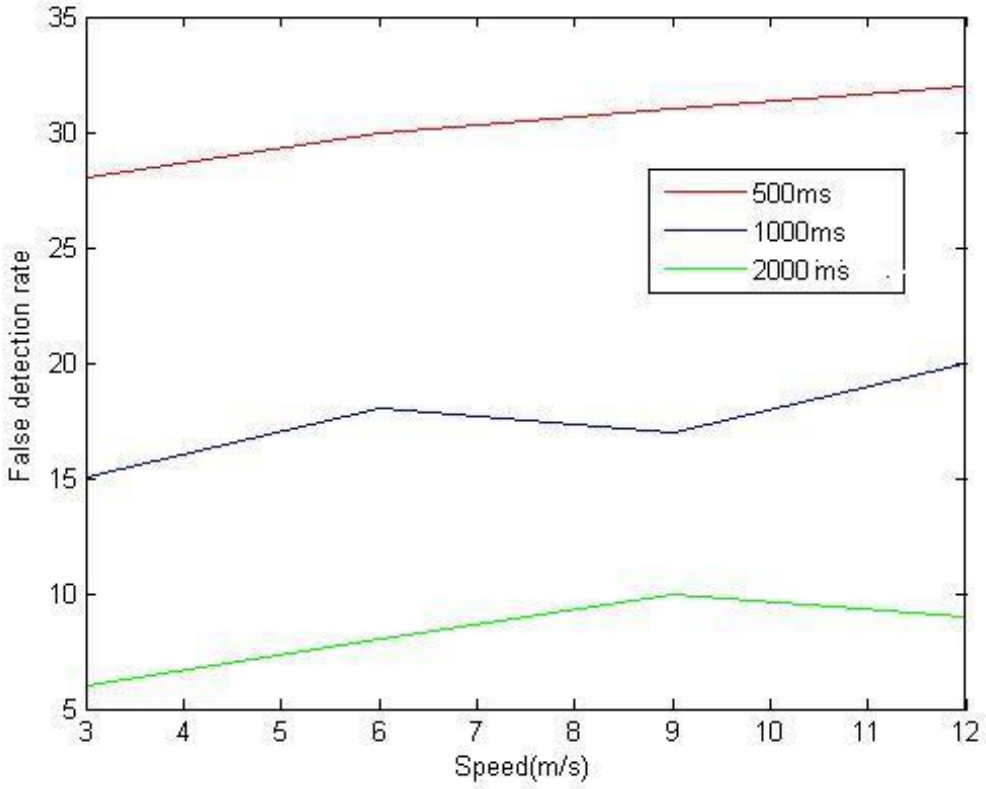

Figure 5.5: FDR of selfish nodes with different action holdoff times

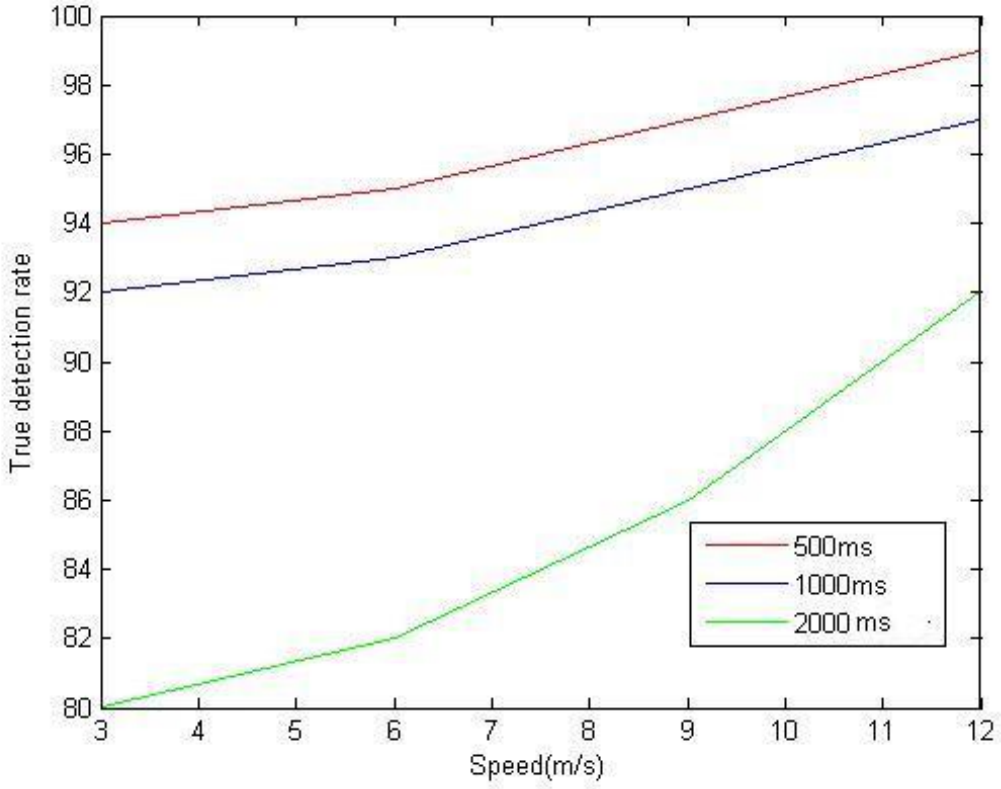

Figure 5.6: TDR of selfish nodes with different action holdoff times

Chapter 6

Chapter 6

Conclusion

6.1 Conclusion

We proposed a time based method for detecting selfish nodes. Sefish nodes in the network do not provide any services to others and reserve resources to itself. Here we proposed a technique for detecting selfish nodes, which don't forward Route Request(RREQ) packets and verified with ns2 simulator, we analyzed the false detection rate, detection rate with different moving rates, different number of selfish nodes in the network and with different action holdoff times, we observed high detection rate when the number of selfish nodes are less and low action holdoff time where as false detection rate is less when the action holdoff time is high.

6.2 Future Work

In this work only nodes which do not send RREQ packets are detected. This work can be extended to detect other types of selfish nodes which can help in improving performence of MANET.

Bibliography

[1] Wu, Lien-Wen, and Rui-Feng Yu. "A threshold-based method for selfish nodes detection in MANET." Computer Symposium (ICS), 2010 International. IEEE, 2010.

[2] Liu, Kejun, et al. "An acknowledgment-based approach for the detection of routing misbehavior in MANETs." Mobile Computing, IEEE Transactions on 6.5 (2007): 536-550.

[3] Kargl, Frank, et al. "Advanced detection of selfish or malicious nodes in ad hoc networks." Security in Ad-hoc and Sensor Networks. Springer Berlin Heidelberg, 2005. 152-165.

[4] Buttyn, Levente, and Jean-Pierre Hubaux. "Stimulating cooperation in self-organizing mobile ad hoc networks." Mobile Networks and Applications 8.5 (2003): 579-592.

[5] Safaei, Zahra, Masoud Sabaei, and Fatemeh Torgheh. "An efficient reputation-based mechanism to enforce cooperation in MANETs." Application of Information and Communication Technologies, 2009. AICT 2009. International Conference on. IEEE, 2009.

[6] Tarannum, Rubana, and Yogadhar Pandey. "Detection and deletion of selfish MANET nodes-a distributed approach." Recent Advances in Information Technology (RAIT), 2012 1st International Conference on. IEEE, 2012.

[7] Samreen, Shirina, and G. Narasimha. "An efficient approach for the detection of node misbehaviour in a MANET based on link misbehaviour." Advance Computing Conference (IACC), 2013 IEEE 3rd International. IEEE, 2013.

[8] Gonzalez, Oscar F., Michael Howarth, and George Pavlou. "An algorithm to detect packet forwarding misbehavior in mobile Ad-Hoc networks." Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on. IEEE, 2007.

[9] Marti, Sergio, et al. "Mitigating routing misbehavior in mobile ad hoc networks." Proceedings of the 6th annual international conference on Mobile computing and networking. ACM, 2000.

[10] Molva, R., and P. Michiardi. "Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks." Institute Eurecom-Research Report RR-02-062 (2001).

[11] Hernandez-Orallo, Enrique, et al. "Improving selfish node detection in MANETs using a collaborative watchdog." Communications Letters, IEEE 16.5 (2012): 642-645.

[12] Fahad, Tarag, and Robert Askwith. "A Node Misbehaviour Detection Mechanism for Mobile Ad-hoc Networks." proceedings of the 7th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting. 2006.

[13] Banerjee, Sukla. "Detection/removal of cooperative black and gray hole attack in mobile ad-hoc networks." Proceedings of the World Congress on Engineering and Computer Science. 2008.

[14] Bakar, Khairul Azmi Abu, and James Irvine. "A Scheme for Detecting Selfish Nodes in MANETs using OMNET++." Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on. IEEE, 2010.

[15] Koshti, Dipali, and Supriya Kamoji. "Comparative study of Techniques used for Detection of Selfish Nodes in Mobile Ad hoc Networks." International Journal of Soft Computing and Engineering (IJSCE) ISSN (2011): 2231-2307.

[16] Das, Samir R., Elizabeth M. Belding-Royer, and Charles E. Perkins. "Ad hoc on-demand distance vector (AODV) routing." (2003).

[17] Roy, Debdutta Barman, and Rituparna Chaki. "Detection of Denial of Service Attack Due to Selfish Node in MANET by Mobile Agent." Recent Trends in Wireless and Mobile Networks. Springer Berlin Heidelberg, 2011. 14-23.

[18] Lin, H., Jos G. Delgado-Frias, and Sirisha Medidi. "Using a cache scheme to detect selfish nodes in mobile ad hoc networks." Communications, Internet, and Information Technology. 2007.

[19] Chakeres, Ian D., and Elizabeth M. Belding-Royer. "AODV routing protocol implementation design." Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on. IEEE, 2004.

[20] Chakeres, Ian D., and Elizabeth M. Belding-Royer. "The utility of hello messages for determining link connectivity." Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on. Vol. 2. IEEE, 2002.

[21] Balakrishnan, Kashyap, Jing Deng, and Pramod K. Varshney. "TWOACK: preventing selfishness in mobile ad hoc networks." Wireless Communications and Networking Conference, 2005 IEEE. Vol. 4. IEEE, 2005.

[22] Buchegger, Sonja, and Jean-Yves Le Boudec. "Performance analysis of the CONFIDANT protocol." Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and computing. ACM, 2002.

[23] Gupta, Shailender, C. K. Nagpal, and Charu Singla. "IMPACT OF SELFISH NODE CONCENTRATION IN MANETS." International Journal of Wireless and Mobile Networks 3.2 (2011).

[24] Vijayan, R., V. Mareeswari, and K. Ramakrishna. "Energy based Trust solution for Detecting Selfish Nodes in MANET using Fuzzy logic." International Journal of Research and Reviews in Computer Science 2.3 (2011).

[25] Wang, Yongwei, Venkata C. Giruka, and Mukesh Singhal. "A fair distributed solution for selfish nodes problem in wireless ad hoc networks." Ad-Hoc, Mobile, and Wireless Networks. Springer Berlin Heidelberg, 2004. 211-224.

[26] Hussain, M. A., et al. "Evaluating network layer selfish behavior and a method to detect and mitigate its effect in MANETs." APA.

[27] Wang, Yongwei, Venkata C. Giruka, and Mukesh Singhal. "A fair distributed solution for selfish nodes problem in wireless ad hoc networks." Ad-Hoc, Mobile, and Wireless Networks. Springer Berlin Heidelberg, 2004. 211-224.

[28] Gupta, Shailender, C. K. Nagpal, and Charu Singla. "IMPACT OF SELFISH NODE CONCENTRATION IN MANETS." International Journal of Wireless and Mobile Networks 3.2 (2011).

[29] Jae Chung, Mark claypool. "Ns By Example." http://nile.wpi.edu/NS/.

[30] Zhibin hu, "Network simulator 2 for wireless". http://www.winlab.rutgers.edu/ zhibinwu/html/network_simulator_2.html.

[31] "Network simulator". http://www.nsnam.com/2011/12/promiscuous-mode-in-aodv-ns-234.html.

[32] Durgesh. "NS-2(2.34) development in Linux." http://durgeshkshirsagar.blogspot.in/2012/07/how-to-use-promiscuous-mode-in-aodv-ns2.html.

[33] Rghu vamsi. http://praghuvamsi.blogspot.in/2013/07/overhearing-packets-or-entering-node.html.

[34] NS-2, The ns Manual (formally known as NS Documentation) available at http: //www. isi.edu/nsnam/ ns/doc.